Automating inductive proof

Automating Inductive Proof

Alan Bundy

School of informatics

University of Edinburgh

Perspectives on Induction

18^{th} July 2014

Practical Importance of Inductive Proof

Needed for reasoning about repetition, e.g.:

- recursive data stuctures;
- recursion and interative programs & plans;
- parameterised hardware,

  e.g., n-bit adder;
- traces of programs,

  e.g., microprocessor, security protocol;

Theoretical Impediments

- Gödel's incompleteness theorems.
- Undecidability of halting problem.
- Lack of cut elimination.

Lack of Cut Elimination

Gentzen's Cut Rule:

$$\frac{\boldsymbol{A}, \Gamma \vdash \Delta, \quad \Gamma \vdash \boldsymbol{A}}{\Gamma \vdash \Delta}$$

lacks subformula property.

Cut Elimination Theorem:

Gentzen showed Cut Rule redundant in FOL.

But necessary in inductive theories.

Practical Consequences:

Need to generalise conjectures.

Need to introduce intermediate lemmas.

Need for non-standard induction rules.

Need for Intermediate Lemmas

Conjecture:

$$\forall l{:}list(\tau).\ rev(rev(l)) = l$$

Rewrite Rules:

$$(nil) <> L \Rightarrow L$$
$$(H :: T) <> L \Rightarrow H :: (T <> L)$$

$$rev(nil) \Rightarrow nil$$
$$rev(H :: T) \Rightarrow rev(T) <> (H :: nil)$$

Upper case indicates meta-variable.

Step Case:

$$rev(rev(t)) = t \vdash rev(\boldsymbol{rev(h :: t)}) = h :: t$$
$$\vdash \underbrace{\boldsymbol{rev(rev(t) <> (h :: nil))}}_{\textbf{blocked}} = h :: t$$

Introducing an Intermediate Lemma

Lemma Required:

$$rev(X <> Y) \Rightarrow rev(Y) <> rev(X)$$

Cut Rule: introduces this:

Original: $\Gamma \vdash rev(rev(l)) = l$

New:

$$\Gamma, \boldsymbol{rev(X <> Y) \Rightarrow rev(Y) <> rev(X)} \vdash rev(rev(l)) = l$$

Justification:

$\Gamma \vdash \boldsymbol{rev(X <> Y) \Rightarrow rev(Y) <> rev(X)}$

Heuristics needed: to speculate lemma.

Step Case Unblocked

$$rev(rev(t)) = t$$

$$\vdash rev(\boldsymbol{rev(h :: t)}) = h :: t$$

$$\vdash \boldsymbol{rev(rev(t) <> (h :: nil))} = h :: t$$

$$\vdash \boldsymbol{rev(h :: nil)} <> rev(rev(t)) = h :: t$$

$$\vdash (\boldsymbol{rev(nil)} <> (h :: nil)) <> rev(rev(t)) = h :: t$$

$$\vdash (\boldsymbol{nil <> (h :: nil)}) <> rev(rev(t)) = h :: t$$

$$\vdash \boldsymbol{(h :: nil) <> rev(rev(t))} = h :: t$$

$$\vdash h :: (\boldsymbol{nil <> rev(rev(t))}) = h :: t$$

$$\vdash \boldsymbol{h :: rev(rev(t)) = h :: t}$$

$$\vdash h = h \wedge \boldsymbol{rev(rev(t)) = t}$$

Now possible to use induction hypothesis.

Rippling in the Step Case

$$t <> (Y <> Z) = (t <> Y) <> Z$$

$$\vdash h :: t^{\uparrow} <> (y <> z) = (h :: t^{\uparrow} <> y) <> z$$

$$\vdash h :: t <> (y <> z)^{\uparrow} = h :: t <> y^{\uparrow} <> z$$

$$\vdash h :: t <> (y <> z)^{\uparrow} = h :: (t <> y) <> z^{\uparrow}$$

$$\vdash h = h \wedge t <> (y <> z) = (t <> y) <> z^{\uparrow}$$

- Changing bits in *orange boxes*$^{\uparrow}$ (wave-fronts).
- Unchanging bits in ***red*** (skeleton).
- Shows embedding of induction hypothesis in induction conclusion.

Wave-Rules

$$H :: \boldsymbol{T}^{\uparrow} \mathbin{<>} \boldsymbol{L} \Rightarrow H :: \boldsymbol{T} \mathbin{<>} \boldsymbol{L}^{\uparrow}$$

$$\boldsymbol{rev}(H :: \boldsymbol{T}^{\uparrow}) \Rightarrow \boldsymbol{rev}(\boldsymbol{T}) \mathbin{<>} (H :: nil)^{\uparrow}$$

$$X_1 :: \boldsymbol{X_2}^{\uparrow} = Y_1 :: \boldsymbol{Y_2}^{\uparrow} \Rightarrow X_1 = Y_1 \wedge \boldsymbol{X_2 = Y_2}^{\uparrow}$$

$$\boldsymbol{X} \mathbin{<>} (\boldsymbol{Y} \mathbin{<>} Z)^{\uparrow} \Rightarrow (\boldsymbol{X} \mathbin{<>} \boldsymbol{Y}) \mathbin{<>} Z^{\uparrow}$$

- Note preservation of skeleton and
- outward movement of wave-fronts.

Rippling Sideways and In

$$rev(t) <> L = qrev(t, L)$$

$$\vdash rev(\ h :: t\ ^{\uparrow}\ ) <> \lfloor l \rfloor = qrev(\ h :: t\ ^{\uparrow}\ , \lfloor l \rfloor)$$

$$\vdash (\ rev(t) <> (h :: nil)\ ^{\uparrow}\ ) <> \lfloor l \rfloor = qrev(t, \ h :: \lfloor l \rfloor\ ^{\downarrow}\ )$$

$$\vdash rev(t) <> (\ (h :: nil) <> \lfloor l \rfloor\ ^{\downarrow}\ ) = qrev(t, \lfloor h :: l \rfloor)$$

$$\vdash rev(t) <> (\lfloor (h :: nil) <> l \rfloor) = qrev(t, \lfloor h :: l \rfloor)$$

$$\vdash rev(t) <> (\lfloor h :: l \rfloor) = qrev(t, \lfloor h :: l \rfloor)$$

- Using induction hypothesis unifies L with $\lfloor h :: l \rfloor$.
- $\lfloor Sinks \rfloor$ provide alternative wave-front destination, available when free variables are in hypothesis.
- Wave-fronts have directions: $out^{\uparrow}$ / $in^{\downarrow}$.
- Note that sinks and wave-fronts may need to be simplified,
  but this is skeleton preserving.

Sideways and Inwards Wave-rules

$$qrev(H :: T^{\uparrow}, L) \Rightarrow qrev(T, H :: L^{\downarrow})$$

$$H :: T <> L^{\downarrow} \Rightarrow H :: T^{\downarrow} <> L$$

$$(X <> Y^{\uparrow}) <> Z \Rightarrow X <> (Y <> Z^{\downarrow})$$

- Note that some equations can be annotated in both directions.

$$H :: T^{\uparrow} <> L \Rightarrow H :: T <> L^{\uparrow}$$

$$H :: T <> L^{\uparrow} \Rightarrow H :: T <> L^{\uparrow}$$

$$X <> (Y <> Z^{\uparrow}) \Rightarrow (X <> Y^{\downarrow}) <> Z$$

Preconditions of the Wave Method

1. The induction conclusion contains a wave-front,

   e.g. $\ldots = \boldsymbol{qrev}(\, h :: \boldsymbol{t}\,^{\uparrow}\, , \lfloor l \rfloor)$.

2. A wave-rule applies to this wave-front.

   e.g. $\boldsymbol{qrev}(\, H :: \boldsymbol{T}\,^{\uparrow}\, , \boldsymbol{L}) \Rightarrow \boldsymbol{qrev}(\boldsymbol{T}, \, H :: \boldsymbol{L}\,^{\downarrow}\,)$.

3. Any condition is provable.

   e.g. $X \neq H \rightarrow \boldsymbol{X} \in H :: \boldsymbol{T}\,^{\uparrow} \Rightarrow \boldsymbol{X} \in \boldsymbol{T}$.

4. Inserted inwards wave-fronts contain a sink or an outwards wave-front.

   e.g. $\ldots = \boldsymbol{qrev}(\boldsymbol{t}, \, h :: \lfloor \boldsymbol{l} \rfloor\,^{\downarrow}\,)$.

Advantages of Rippling

Selective: not exhaustive rewriting.

skeleton preserving and measure decreasing.

Bi-directional: rewriting.

different annotations in each direction.

Termination: of any set of wave-rules,

despite bi-directionality.

Heuristic basis: for choosing lemmas,

generalisations, case splits and inductions.

Ripple-Based Heuristics

Induction Rules: choose induction which best supports rippling.

Lemmas: design wave-rule to unblock ripple.

Generalisation: generalise goal to allow wave-rule to apply.

Critic: Lemma Speculation

Conjecture:

$$rev(rev(L)) = L$$

Wave-Rule:

$$rev(\ H :: T\ ^{\uparrow}\ ) \Rightarrow rev(T) <> H :: nil\ ^{\uparrow}$$

Induction Conclusion:

$$rev(rev(\ h :: t\ ^{\uparrow}\ )) = h :: t\ ^{\uparrow}$$

$$\underbrace{rev(\ rev(t) <> (h :: nil)\ ^{\uparrow}\ )}_{\textbf{blocked}} = h :: t\ ^{\uparrow}$$

Pattern Sought:

$$rev(\ X <> Y\ ^{\uparrow}\ ) \Rightarrow F(rev(X), X, Y)\ ^{\uparrow}$$

Lemma Discovered:

$$rev(\ X <> Y\ ^{\uparrow}\ ) \Rightarrow rev(Y) <> rev(X)\ ^{\uparrow}$$

Failure of Ripple Precondition

- Precondition 1 is true:
  1. The induction conclusion contains a wave-front.

$$rev(\boxed{rev(t) <> (h :: nil)}^{\uparrow}) = \boxed{h :: t}^{\uparrow}$$

     (in fact, two)

- Precondition 2 is false:
  2. A wave-rule applies to this wave-front.
     (to neither of them)
- Preconditions 3 and 4 are inapplicable.
  3. Any condition is provable.
  4. Inserted inwards wave-fronts contain a sink or an outwards wave-front.

Rippling Failure: Missing Sink

Conjecture:

$$\forall t{:}list(\tau).\ rev(t) = qrev(t, nil)$$

Wave-Rules:

$$rev(\ H :: \boxed{T}^{\uparrow}\ ) \Rightarrow \boxed{rev(T)} <> H :: nil^{\uparrow}$$

$$qrev(\ H :: \boxed{T}^{\uparrow}, L) \Rightarrow qrev(T, \boxed{H :: L}^{\downarrow}\ )$$

Induction Conclusion:

$$rev(\ h :: \boxed{t}^{\uparrow}\ ) = qrev(\ h :: \boxed{t}^{\uparrow}, nil)$$

$$\boxed{rev(t)} <> h :: nil^{\uparrow} = \underbrace{qrev(\ h :: \boxed{t}^{\uparrow}, nil)}_{\textbf{missing sink}}$$

Failure of Ripple Precondition

- Preconditions 1, 2 and 3 are true:
  1. The induction conclusion contains a wave-front.

$$\ldots = \boldsymbol{qrev}(\, h :: \boldsymbol{t}\,^{\uparrow}, \boldsymbol{nil})$$

  2. A wave-rule applies to this wave-front.

$$\boldsymbol{qrev}(\, H :: \boldsymbol{T}\,^{\uparrow}, \boldsymbol{L}) \Rightarrow \boldsymbol{qrev}(\boldsymbol{T}, \, H :: \boldsymbol{L}\,^{\downarrow})$$

  3. Any condition is provable — trivially, no condition.

- Precondition 4 is false:
  4. Inserted inwards wave-fronts contain a sink or an outwards wave-front.

$$\ldots = \boldsymbol{qrev}(\boldsymbol{t}, \, h :: \boldsymbol{nil}\,^{\downarrow})$$

Patch: Sink Speculation

Original Conjecture:

$$\forall t{:}list(\tau).\ rev(t) = qrev(t, nil)$$

Disallowed Ripple:

$$\ldots = \boldsymbol{qrev(t,}\ h :: \boldsymbol{nil}^{\downarrow}\ \boldsymbol{)}$$

Schematic Conjecture:

$$\forall t{:}list(\tau).\forall l{:}list(\tau).\ F(rev(t), l) = qrev(t, G(l))$$

Induction Hypothesis:

$$\boldsymbol{F(rev(t), L) = qrev(t, G(L))}$$

where F, G and L are meta-variables.

Patch: Instantiating the Meta-Variables

New Step Case:

$$F(rev(h :: t)^{\uparrow}, \lfloor l \rfloor) = qrev(h :: t)^{\uparrow}, G(\lfloor l \rfloor))$$

$$F(rev(t) <> h :: nil)^{\uparrow}, \lfloor l \rfloor) = qrev(t, h :: G(\lfloor l \rfloor))^{\downarrow})$$

$$rev(t) <> (h :: nil <> F'(rev(t) <> (h :: nil))^{\uparrow}, \lfloor l \rfloor))^{\downarrow}$$

$$= qrev(t, h :: G(\lfloor l \rfloor))^{\downarrow})$$

$$rev(t) <> (h :: F'(rev(t) <> (h :: nil))^{\uparrow}, \lfloor l \rfloor))^{\downarrow}$$

$$= qrev(t, h :: G(\lfloor l \rfloor))^{\downarrow})$$

$$rev(t) <> (\lfloor h :: l \rfloor) = qrev(t, \lfloor h :: l \rfloor)$$

where $F = <>$, $F' = \lambda X.\lambda Y.\ Y$ and $G = \lambda X.\ X$.

Key Wave-Rule: $(X <> Y)^{\uparrow} <> Z \Rightarrow X <> (Y <> Z)^{\downarrow}$

Generalised Conjecture:

$\forall t{:}list(\tau).\forall l{:}list(\tau).\ rev(t) <> l = qrev(t, l)$

Pattern of Failure Suggests Patch

	PC 1	PC 2	PC 3	PC 4
Generalization	$\surd$	$\surd$	$\surd$	$\times$
Case Split	$\surd$	$\surd$	$\times$	
Induction Revision	$\surd$	?		
Lemma Discovery	$\surd$	$\times$		

$\surd$ = success ? = partial success $\times$ = failure

Conclusion

- Negative theoretical results create special search problems,

  especially lack of cut elimination.
- These problems common in practice:

  induction rule choice, lemmas & generalisations.
- Inductive step case guided by rippling.
- Rippling: selective; bidirectional; terminating and

  offers heuristic choice of cut formula.
- Ripple breakdowns suggest: induction revision; lemma speculation or generalisation.

  Different patterns of proof breakdown suggest different patches.
- Implemented via proof planning with critics.